

KOREA 2020

DISCOVERY  
SUMMIT

ONLINE



Copyright © SAS Institute Inc. All rights reserved.

# Data pipeline construction and analysis automation using JMP & Python

INNOVALUE  
PARTNERS

Jong-Chan Kim



# Presenter



## Jong – Chan Kim

- Senior Researcher, InnoValue Partners
  - AI & BD Camp Data Analysis Instructor
  - POSCO HRD Professor
- ✓ Big data analysis education consulting using JMP
  - ✓ Data analysis education consulting using Python/R
  - ✓ Artificial Neural Network & Deep Learning (Artificial Neural Network & Deep Learning)
  - ✓ Python crawling and data analysis platform construction

KOREA 2020

DISCOVERY  
SUMMIT  
ONLINE

Copyright © SAS Institute Inc. All rights reserved.

jmp

# Data Analysis Process



Determining the four basic steps of the data analysis process, collection, storage, processing and analysis is very important to business.

And by using a tool suitable for each step, you need to analyze data quickly and easily.

# JMP Program

- JMP is very powerful and suitable for importing data, visualizing it and performing statistical analysis.
- JMP is a user-friendly and easy-to-use UI that can be easily used by users who want to analyze data.
- However, the easy-to-use UI has clear limitations.

# JMP Script Language, JSL

- In order for users to process and analyze data faster and more automatically, they need the help of Script.
- If you use JMP Script, you can process and analyze data more powerfully and quickly, and you can build automatic procedures.
- And by combining the Python programming language in JMP Script, you can enjoy this effect even more.

# Python

- Python is object-oriented.
- Python can use several libraries.
- Python is simple and suitable for data analysis.

\*\*One thing to note is that when using Python in JMP, check the Version carefully.

# Using Python in JMP

Version of whether the Python programming language is available in JMP  
Confirm.

```
ver = Trim(  
  Run Program(  
    Executable( "python.exe" ),  
    Options( "--version" ),  
    Read Function( "text" )  
  )  
);  
  
Show( ver );
```



# Using Python in JMP

Calls a Python program and performs calculations using Python.

```
Python Init();  
Python Submit( "df2= 100+200" );  
Python Submit("print(df2)");  
Python Term();
```

# Using Python in JMP

Load one example script file and declare it as a variable.

```
Python Init();  
df1 = Open("$SAMPLE_DATA/Big Class.jmp", invisible );  
Python Send(df1);
```

# Using Python in JMP

Process the data, using Python syntax.

The functions below are the functions to grasp basic property information of data.

It uses a Python library called Pandas

# Using Python in JMP

Check only the top 6 data. Then, summarizes the continuous data and saves the data under the name 'result.csv'

```
Python Submit("\n[\nimport pandas as pd\n\nprint(df1.head())\n\nprint(df1.describe())\n\ndf2 = df1.pivot_table(index='sex', values='height', aggfunc='sum').reset_index()\n\ndf2.to_csv('/Users/ainbd/Desktop/result.csv')\n\n]\n");
```

# Using Python in JMP

```
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
0
```

	name	age	sex	height	weight
0	KATIE	12	F	59	95
1	LOUISE	12	F	61	123
2	JANE	12	F	55	74
3	JACLYN	12	F	66	145
4	LILLIE	12	F	52	64

	age	height	weight
count	40.000000	40.000000	40.000000
mean	13.975000	62.550000	105.000000
std	1.476092	4.242338	22.201871
min	12.000000	51.000000	64.000000
25%	13.000000	60.750000	91.750000
50%	14.000000	63.000000	105.000000
75%	15.000000	65.000000	115.250000
max	17.000000	70.000000	172.000000

KOREA 2020

DISCOVERY  
SUMMIT  
ONLINE

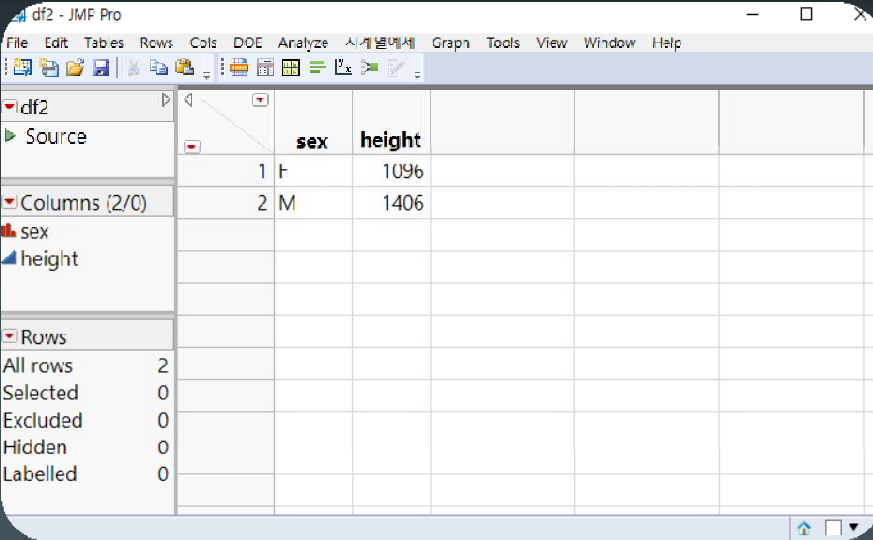
Copyright © SAS Institute Inc. All rights reserved.

jmp

# Using Python in JMP

The result of the operation calculated in Python is brought back to JMP and a window is displayed.

```
df2 = Python Get( df2 );  
df2 << New Data View;  
Python Term();
```



The screenshot shows the JMP Pro interface with a data table. The table has two columns: 'sex' and 'height'. The first row shows '1 F' and '1096', and the second row shows '2 M' and '1406'. The interface also shows a menu bar, a toolbar, and a sidebar with a tree view of the data table.

	sex	height
1	F	1096
2	M	1406

KOREA 2020

DISCOVERY  
SUMMIT  
ONLINE

Copyright © SAS Institute Inc. All rights reserved.

jmp

# Using Python in JMP

If you use this well, you can easily and simply automate tasks with Python, and visualize them with JMP.

Let's look at an example of a simple automation of repetitive daily tasks in the actual retail industry.

# E-commerce Case

The company needed to calculate and visualize the delivery and inventory of certain items every week.

날짜	SKU ID	SKU 명	비고	발주가능상태	입고수량	구매준고수량	구매원재재고수량	자재번호	자재명	기대고리
1 2019-05-16	29349	리뷰진단결...	880104689...	발주가능	384	287	5788	8077374	19리규...	세탁세제
2 2019-05-16	29350	리규...	880104689...	발주가능	0	285	2772	8077394	19리규...	세탁세제
3 2019-05-16	29351	울산푸아웃...	880104687...	발주가능	0	2	35	8077406	19울산푸아...	세탁세제
4 2019-05-16	29352	울산푸아웃...	880104687...	발주가능	207	41	2083	8077404	19울산푸아...	세탁세제
5 2019-05-16	29359	헤어케어...	880104688...	발주가능	288	312	3411	5009109	182080우...	헤어케어
6 2019-05-16	32052	케어시스...	880104699...	발주가능	0	0	88	8075349	19KS피플...	헤어케어
7 2019-05-16	32053	케어시스...	880104699...	발주가능	0	662	10164	8075350	19KS피플...	헤어케어
8 2019-05-16	32054	케어시스...	880104699...	발주가능	520	190	6709	8075351	19KS피플...	헤어케어
9 2019-05-16	32055	케어시스...	880104699...	발주가능	80	88	1071	8075352	19KS피플...	헤어케어
10 2019-05-16	34359	리뷰진단결...	880104624...	발주가능	0	82	2689	8078472	19리규...	세탁세제
11 2019-05-16	34402	울산푸아웃...	880104687...	발주가능	1120	276	6074	8068101	18울산푸아...	세탁세제
12 2019-05-16	34403	울산푸아...	880104687...	발주가능	0	60	2464	8067114	18울산푸아...	세탁세제
13 2019-05-16	34404	울산푸아웃...	880104624...	발주가능	0	2	0	8067119	18울산푸아...	세탁세제
14 2019-05-16	41345	리규...	880104624...	발주가능	0	6	455	8067081	17리규미...	세탁세제
15 2019-05-16	41346	리규...	880104624...	발주가능	96	12	323	8067082	17리규미...	세탁세제
16 2019-05-16	41357	얼마의선택...	880104624...	발주가능	816	147	4505	8075861	19얼마의선택...	세탁세제
17 2019-05-16	41358	순삼Bubble...	880104689...	발주가능	0	88	1187	8057471	10순삼Bub...	수방세제
18 2019-05-16	45232	케어시스...	880104624...	발주가능	0	44	321	8075574	19KS핀스...	헤어케어
19 2019-05-16	45237	케어시스...	880104624...	발주가능	120	151	887	8075906	19KS핀스...	헤어케어

KOREA 2020



Copyright © SAS Institute Inc. All rights reserved.





# E-commerce Case

The process was automated using several Python data preprocessing techniques and JMP visualization.

KOREA 2020

DISCOVERY  
SUMMIT  
ONLINE

Copyright © SAS Institute Inc. All rights reserved.

jmp

# E-commerce Case

```
Python Submit( "\[
import pandas as pd

print(df1.head())

print(df1.describe())

cond1 = (df1['카테고리']=='세탁세제')
df2 = df1.loc[cond1]

df2['datetime'] = pd.to_datetime(df2['날짜'])

df2['year'] = df2['datetime'].dt.year
df2['week'] = df2['datetime'].dt.week
df2['day'] = df2['datetime'].dt.day

df3 = df2[['datetime','카테고리','출고수량','재고수량','year','week','day']]

]\n );

df3 = Python Get( df3 );
```

KOREA 2020

DISCOVERY  
SUMMIT  
ONLINE

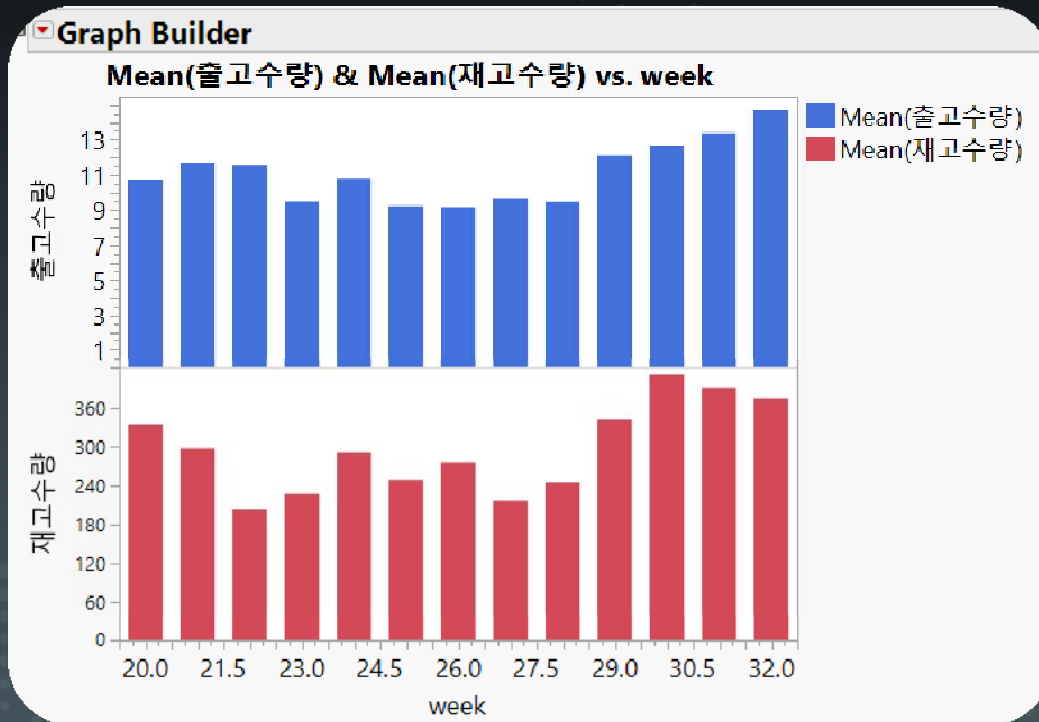
Copyright © SAS Institute Inc. All rights reserved.

jmp

# E-commerce Case

```
df3 << New Data View;  
  
Graph Builder(df3,  
  Size( 518, 448 ),  
  Show Control Panel( 0 ),  
  Variables( X( :week ), Y( :출고수량 ), Y( :재고수량 ) ),  
  Elements( Position( 1, 1 ), Bar( X, Y, Legend( 18 ) ) ),  
  Elements( Position( 1, 2 ), Bar( X, Y, Legend( 19 ) ) )  
);
```

# E-commerce Case



KOREA 2020

DISCOVERY  
SUMMIT  
ONLINE

Copyright © SAS Institute Inc. All rights reserved.

jmp

KOREA 2020

DISCOVERY  
SUMMIT  
ONLINE

0001jmp@gmail.com



Copyright © SAS Institute Inc. All rights reserved.